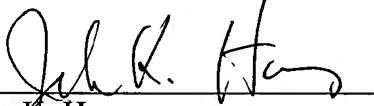


IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF E-FILING

I hereby certify that this correspondence has been e-filed with the U.S. Patent and Trademark Office on April 8, 2010.



John K. Harrop
Reg. No. 41,817

Appl. No. : 10/690,605
Applicant : Sundaresan Ramamoorthy, et al.
Filed : October 23, 2003
Title : SMART TRANSLATION OF GENERIC CONFIGURATION
TC/A.U. : 2436
Examiner : Daniel L. Hoang
Docket No. : 200207938-1
Customer No. : 022879

Mail Stop Appeal Brief - Patents
Commissioner of Patents
P.O. Box 1450
Alexandria, Virginia 22313-1450

APPEAL BRIEF UNDER 37 C.F.R. §41.37

TABLE OF CONTENTS

	<u>Page</u>
I. REAL PARTY IN INTEREST	3
II. RELATED APPEALS AND INTERFERENCES	4
III. STATUS OF CLAIMS	5
IV. STATUS OF AMENDMENTS	6
V. SUMMARY OF CLAIMED SUBJECT MATTER.....	7
VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL	9
VII. ARGUMENT.....	10
VIII. CLAIMS - APPENDIX.....	15
IX. EVIDENCE – APPENDIX.....	19
X. RELATED PROCEEDINGS – APPENDIX	20

I. REAL PARTY IN INTEREST

Hewlett Packard Company is the real party in interest.

II. RELATED APPEALS AND INTERFERENCES

There are no other related appeals or interferences.

III. STATUS OF CLAIMS

Claims 1 – 4 and 6 – 18 are pending. Claim 5 is cancelled. Claims 1 – 4 and 6 – 18 are rejected under 35 U.S.C. §102(b) over U.S. Patent Publication 20040193912 to Whipple. Appellants appeal the rejection of claims 1 – 4 and 6 – 18.

IV. STATUS OF AMENDMENTS

There are no amendments after final that have not been entered.

V. SUMMARY OF CLAIMED SUBJECT MATTER

The inventions are recited in three independent claims: claims 1, 10, and 18, and their associated dependent claims. Claim 1 is best understood with reference to Figure 1, and its accompanying description at page 4, lines 7 – 23, and 30 – 33, and at page 5, line 1 through page 7, line 26. Claim 1 recites a system for implementing a policy in a network 100. The system includes a plurality of device-agnostic policy implementations 120, which include non-security policy implementations (i.e., other than 122)(page 3, lines 9 – 16). The system also includes a plurality of network devices 140, and at least two of the network devices 140 are dissimilar. A type of network device 140 associated with a received device-agnostic policy implementation 120 is identified by parsing tags of data from the received device-agnostic policy implementation 120 (page 3, lines 23 – 26) represented using Extensible Markup Language (XML)(page 3, lines 4 – 7). The system further includes a plurality of device translators 130, and each device translator 130 corresponds to a respective one of the network devices 140 and to one of the device-agnostic policy implementations 120 (page 3, lines 21 – 23). At least two of the device translators 130 are dissimilar. Each of the device translators 130 translates the device-agnostic policy implementation 120 into a corresponding device-specific implementation (page 3, lines 21 – 23). Subsequent additions or maintenance of any of the network device-agnostic policy implementations 120 are provided using device-agnostic files (page 4, lines 30 – 33). A specific example of the system of claim 1 is illustrated starting at page 5, line 29 and ending at page 6, line 28, in which a XML file (i.e., a device agnostic policy implementation 120) for use with a firewall is translated, using an XSL translator (i.e., translator 130) to generate a JAVA file appropriate for a specific network device, namely a Cisco PIX (i.e., network device 140).

Claim 10 is best understood with reference to Figures 1 and 2 and their accompanying description at page 4, lines 7 - 29 and page 5, line 1 through page 7, line 26. Claim 10 recites a computer-implemented method including the steps of representing 205 a vendor-agnostic configuration 120 using a processor in a computer connected to a computer network 100, building 210 a translator 130, using the processor, for a specific policy and vendor, in which the computer network 100 includes a plurality of policies and vendors, the policies including non-security policies, repeats the building 210 for each type of policy and vendor, identifying 215 a type of device 140 by parsing tags of data from the received vendor-agnostic configuration represented using Extensible Markup Language (XML)(page 3, lines 23 – 26),

loading 220 the translator 130 into memory, translating 225 the vendor-agnostic configuration 120 into a vendor-specific configuration using the translator 130 and repeating the identifying, loading and translating for each type of policy and vendor.

Computer readable medium claim 18 is best understood with reference to Figures 1 and 2 and their accompanying descriptions at page 4, lines 7 - 33 and page 5, line 1 through page 7, line 26. Claim 18 recites instructions for implementing a policy in a computer network 100. The instructions include representing 205 a vendor-agnostic configuration 120, building 210 a translator 130 for a specific policy and vendor, in which the computer network 100 includes a plurality of policies and vendors, the policies including non-security policies, repeating the building 210 for each type of policy and vendor, identifying 215 a device 140, loading 220 the translator 130 after identifying the type of device 140, translating 225 the vendor-agnostic configuration 120 into a vendor-specific configuration using the translator 130 (page 4, lines 24 – 29; Figure 2), repeating the identifying, loading and translating steps for each type of policy and vendor, and providing subsequent additions or maintenance of any policies and vendors using device-agnostic files (page 4, lines 30 – 33).

VI. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Claims 1 – 4 and 6 – 18 are rejected under 35 U.S.C. §102(b) over U.S. Patent Publication 2002/0038340 to Whipple et al.

VII. ARGUMENT

Claims 1 – 4 and 6 – 18 are rejected under 35 U.S.C. §102(b) over U.S. Patent Publication 2002/0038340 to Whipple et al. (hereafter Whipple). Claim 1 is a system claim; independent claim 10 is a method claim; and independent claim 18 is a computer-readable medium claim.

“A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently described, in a single prior art reference,” Verdegaal Bros. v. Union Oil Co. of California, 814 F.2d 628, 631, 2 USPQ2d 1051, 1053 (Fed. Cir. 1987); see also MPEP § 2131. “The identical invention must be shown in as complete detail as is contained in the ... claim.” Richardson v. Suzuki Motor Co., 9 USPQ2d 1913, 1920 (Fed. Cir. 1989). The elements must be arranged as required by the claim. In re Bond, 15 USPQ2d 1566 (Fed. Cir. 1990). Whipple does not show each and every element as set forth in the claims, does not suggest the identical invention as defined in the claims, and does not suggest the elements arranged as required by the claims.

A. Whipple Does Not Disclose or Suggest Any Element of the Independent Claim 1

Considering claim 1, Whipple does not describe, disclose, or suggest “wherein *a type of network device* associated with a received device-agnostic policy implementation *is identified by parsing tags of data* from said received device-agnostic policy implementation represented using Extensible Markup Language (XML) ... each of said plurality of device translators being loaded *after* said type of network device is identified to *translate said received device-agnostic policy implementation into corresponding device-specific implementations*,” as recited in claim 1 (emphasis added).

1. Claim 1 Recites Translation from Device-Agnostic to Device Specific; Whipple Discloses Translation from Device-Specific to Device-Specific

In an April 2, 2009 final Office Action, the Examiner states “Applicant’s invention, as far as examiner can understand, relates to a technique for which vendor-specific tools can be translated to a single vendor-agnostic configuration. For example, when a vendor specific tool which uses a vendor specific format of XSL wishes to communicate with a vendor using an XML format, a translator is built and loaded in order to handle communication (see

paragraph 20 of the instant application).” In fact, the examiner’s statement does not reflect the claimed invention.

Claim 1 recites, in the first element, a device agnostic-policy implementation. An example of such a device-agnostic policy implementation is the XML code string (or file) starting on page 5, line 31 (<policy type= “security”), and ending at page 6, line 5. The second element of claim 1 is a network device. An example of such a network device is a Cisco PIX firewall (see page 5, line 30). (A Cisco PIX (Private Internet eXchange) is an IP firewall and network address translation (NAT) appliance.) To make the device-agnostic policy implementation represented by the XML file device-specific, that policy implementation (file) is translated using a translator, which is the third element of claim 1. In the example bridging pages 5 and 6, the translator is an XSL file, shown on page 6, lines 10 – 15. Thus, the XSL file extracts relevant information, by first parsing the XML file to identify the appropriate specific network device (in the example, a Cisco PIX firewall) and then using the XSL file to translate the XML file into code (specifically an API) appropriate for the Cisco PIX firewall. The XSL file translates the XML code to a Java API.

That the invention recited in claim 1 does indeed translate from device-agnostic to device-specific can be seen with reference to the second code translation example, which is provided at page 6, line 28 to page 7, line 19. In this example, the translation is for a Load Balancer firewall. Simple inspection of the starting XML files used in the two examples shows they are identical. That is, compare the XML file from page 5, line 31 to page 6, line 5 with the XML file from page 6, line 31 to page 7, line 4. The two file are identical – that is, the XML file is device-agnostic. The XSL translator also is identical in both examples, as would be expected. But the resulting translated files, that is, the device-specific implementations, are not identical. They are not identical because the translated files apply to different network devices (both firewalls, but from different manufacturers, Cisco and Load Balancer).

Thus, a careful reading of claim 1, as clearly supported in the specification, shows that the examiner may have mis-read the claimed invention. Since the examiner appears confused about the exact nature of the invention, one might expect that he would not apply an appropriate reference in rejecting the claim, or, alternatively, would mis-apply an appropriate reference. In either event, the application of Whipple to claim 1 is flawed for at least the fact

that claim 1 recites translation from device-agnostic to device-specific, while Whipple, as will be explained below, translates a device-specific API to another device-specific API.

Whipple is directed to a method and system for translating “API formats used by clients 18 to a format appropriate for [a] hub API.” See paragraph 0016. As Whipple points out in paragraphs 0015 – 0016, the clients 18 are, or use, distinct or specific devices. The APIs employed by the clients 18 are therefore not “device-agnostic,” but rather are device-specific. However, the APIs used by the clients 18 may not conform to the API used by hub 12. Thus, Whipple invokes a translation scheme, described, for example, in paragraph 0023, to convert the client’s API to a “representation suitable for processing at the hub application server.” This translation may involve translation of XML to Java, but the translation is most definitely not from a device-agnostic API to a device-specific API.

2. Claim 1 Recites Parsing Tags; Whipple Discloses Reading a Wrapper; The Two Are Not the Same Or Equivalent

In a “Response to Arguments” made in the December 24, 2009 final Office Action, the Examiner “interprets” Whipple’s reading “of the wrapper that contains the NAPI request” as “analogous to the [claimed] parsing of tags,” and that Whipple’s selection “of an appropriate adapter” is “analogous [to] the applicant’s claimed ‘identifying the network device.’” More specifically, the final Office Action argues that Whipple teaches “the network API request component including a description of a system API method to be called and one or more parameters to be used in executing the system API” and argues that this is analogous to identifying the network device associated with a received device-agnostic policy implementation by parsing tags of data from received device-agnostic policy implementation.

First, as noted above, for a claim to be anticipated, the *identical* invention must be shown in as complete detail as is contained in the claim. Richardson v. Suzuki Motor Co., 9 USPQ2d 1913, 1920 (Fed. Cir. 1989). Whipple does not show parsing of tags of data, as the examiner admits (analogous is not identical), and therefore, does not show the identical invention in as complete detail as is claimed.

Second, what Whipple does disclose, reading a “wrapper that contains the NAPI request,” is not even analogous, or equivalent, to parsing tags. By reading the wrapper, Whipple’s system is able to determine the format of the associated a request (i.e., determine if the request is in XML or Java, formats, for example). However, knowing the format of the

request does not mean Whipple's system has access to, or knows the content of the specific request. So if Whipple's system determines that the request is in XML, and Whipple's hub application server uses Java, Whipple's system selects a XML-Java adapter 24 to translate from the client, device-specific API, to the hub system's API, which also is device-specific (i.e., specific to the hub application server 32)(see paragraph 0023).

Third, the description of a system API method to be called and parameters to be used are not an identification of "the network device associated with a received device-agnostic policy implementation." The received "policy implementation" in Whipple is device-specific, not device-agnostic. Moreover, "the device associated with the received ... policy implementation" is the client 18. Nowhere does Whipple disclose (or suggest) that the client 18 is identified.

Fourth, the Office Action has failed to identify what in the network API request component is an identification of the network device associated with a received device-agnostic policy implementation. Whipple simply fails to describe this feature.

Thus, Whipple's system and method are neither identical to nor "analogous to the [claimed] parsing of tags," and Whipple's selection "of an appropriate adapter" is neither identical to nor "analogous [to] the applicant's claimed 'identifying the network device.'"

3. Whipple Does Not Disclose a Device Translator Corresponding to a Network Device

Applicants maintain their prior argument that Whipple does not describe "each device translator corresponding to a respective one of said plurality of network devices." That Whipple teaches that each format preferably has a corresponding API adapter, as asserted on page 3 Office Action, does not disclose that each device translator corresponds to one network device.

For at least the reasons stated in paragraphs A.1 – A.3, above, Whipple does not anticipate claim 1, and claim 1 is patentable.

B. Claims 2 - 18

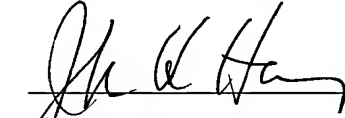
Independent claims 10 and 18 recite elements that are the same as, or similar to, those in patentable claim 1. Accordingly, claims 10 and 18 also are patentable. Dependent claims 2 – 9 and 11 – 17 also are patentable for these reasons and for the elements they recite.

The appeal brief fee in the amount of \$540.00 was previously paid. Filed herewith is a petition for extension of time of one (1) month. Please charge the extension of time fee to **Deposit Account 20-0823**. Please charge any additional fees required for this appeal brief, please charge any fees or credit any over payment to **Deposit Account 08-2025** pursuant to 37 C.F.R. §1.25.

Date: _____

April 8, 2010

Respectfully submitted,



John K. Harrop
Registration No. 41817
Thompson Coburn LLP
1909 K Street, NW
Suite 600
Washington, DC 20006
Tel. (202) 585-6979
Fax. (202) 508-1038

VIII. CLAIMS - APPENDIX

Claim 1: A system for implementing a policy in a network, said system comprising:

a plurality of device-agnostic policy implementations, in which the device-agnostic policy implementations include non-security policy implementations;

a plurality of network devices, at least two of said devices being dissimilar, wherein a type of network device associated with a received device-agnostic policy implementation is identified by parsing tags of data from said received device-agnostic policy implementation represented using Extensible Markup Language (XML); and

a plurality of device translators, each device translator corresponding to a respective one of said plurality of network devices and one of said plurality of device-agnostic policy implementations, at least two of said device translators being dissimilar, each of said plurality of device translators translating said device-agnostic policy implementation into corresponding device-specific implementations, wherein subsequent additions or maintenance of any of said plurality of network device-agnostic policy implementations are provided using device-agnostic files.

Claim 2: The system according to claim 1, wherein said device-agnostic policy implementation is selected from the group consisting of firewall, Virtual Private Network, Java 2 Enterprise Edition Application, and custom operating system.

Claim 3: The system according to claim 1, wherein said device-agnostic policy implementation implements a policy selected from the group consisting of access control, quality of service, backup, and availability.

Claim 4: The system according to claim 1, wherein said translators are represented by Extensible Stylesheet Language (XSL) code.

Claim 5: cancelled.

Claim 6: The system according to claim 3, wherein said policy is represented by Extensible Markup Language (XML) code.

Claim 7: The system according to claim 1, wherein the device specific implementation is represented by Command Line Interface (CLI) code.

Claim 8: The system according to claim 1, wherein the device-specific implementation is represented by Application Programming Interface (API) code.

Claim 9: The system according to claim 1, wherein the device-specific implementation is represented by Java code.

Claim 10: A computer-implemented method comprising:

representing a vendor-agnostic configuration using a processor in a computer connected to a computer network;

building a translator, using the processor, for a specific policy and vendor, in which the computer network includes a plurality of policies and vendors, the policies including non-security policies;

repeating the building for each type of policy and vendor;

identifying a type of device associated with a received vendor-agnostic configuration by parsing tags of data from said received vendor-agnostic configuration represented using Extensible Markup Language (XML), using the processor;

loading said translator into memory, using the processor, after identifying said type of device;

translating said vendor-agnostic configuration into vendor-specific configuration using said translator; and

repeating the identifying, loading and translating for each type of policy and vendor; and

providing subsequent additions or maintenance of any of said plurality of policies and vendors using device-agnostic files.

Claim 11: The method according to claim 10, wherein said vendor-agnostic configuration is represented by Extensible Markup Language (XML) code.

Claim 12: The method according to claim 10, wherein said translator is represented by Extensible Stylesheet Language (XSL) code.

Claim 13: The system according to claim 10, wherein said specific policy is selected from the group consisting of firewall, Virtual Private Network, Java 2 Enterprise Edition Application, and custom operating system.

Claim 14: The system according to claim 10, wherein said specific policy is selected from the group consisting of access control, quality of service, backup, and availability.

Claim 15: The system according to claim 10, wherein the vendor-specific configuration is represented by Command Line Interface (CLI) code.

Claim 16: The system according to claim 10, wherein the vendor-specific configuration is represented by Application Programming Interface (API) code.

Claim 17: The system according to claim 10, wherein the vendor-specific configuration is represented by Java code.

Claim 18: A computer readable medium containing instructions for implementing a policy in a computer network, said instructions comprising:

representing a vendor-agnostic configuration;

building a translator for a specific policy and a specific vendor, in which the computer network includes a plurality of policies and vendors, the policies including non-security policies;

repeating the building for each type of policy and vendor;

identifying a type of device associated with a received vendor-agnostic configuration by parsing tags of data from said received vendor-agnostic configuration represented using Extensible Markup Language (XML);

loading said translator after identifying said type of device;

translating said received vendor-agnostic configuration into vendor-specific configuration using said translator;

repeating the identifying, loading and translating for each type of policy and vendor;

and

providing subsequent additions or maintenance of any said plurality of policies and vendors using device-agnostic files.

IX. EVIDENCE – APPENDIX

No evidence is submitted.

X. RELATED PROCEEDINGS – APPENDIX

There are no related proceedings.